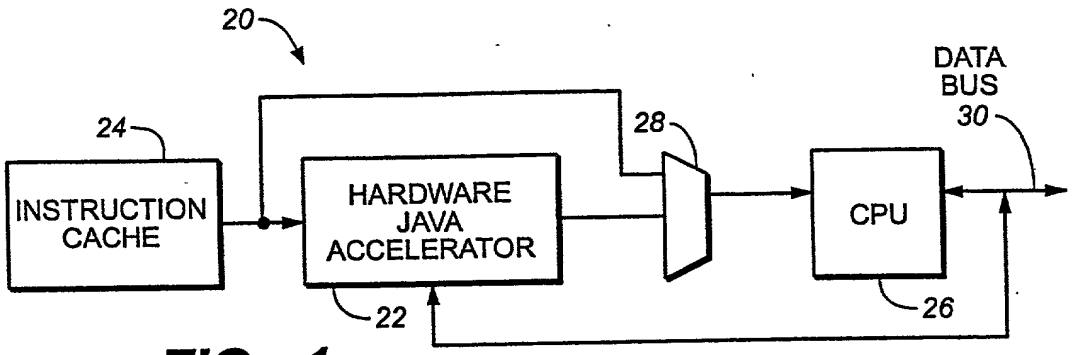
20
24
INSTRUCTION CACHE
HARDWARE JAVA ACCELERATOR
22
28
DATA BUS
30
CPU
26

FIG._1

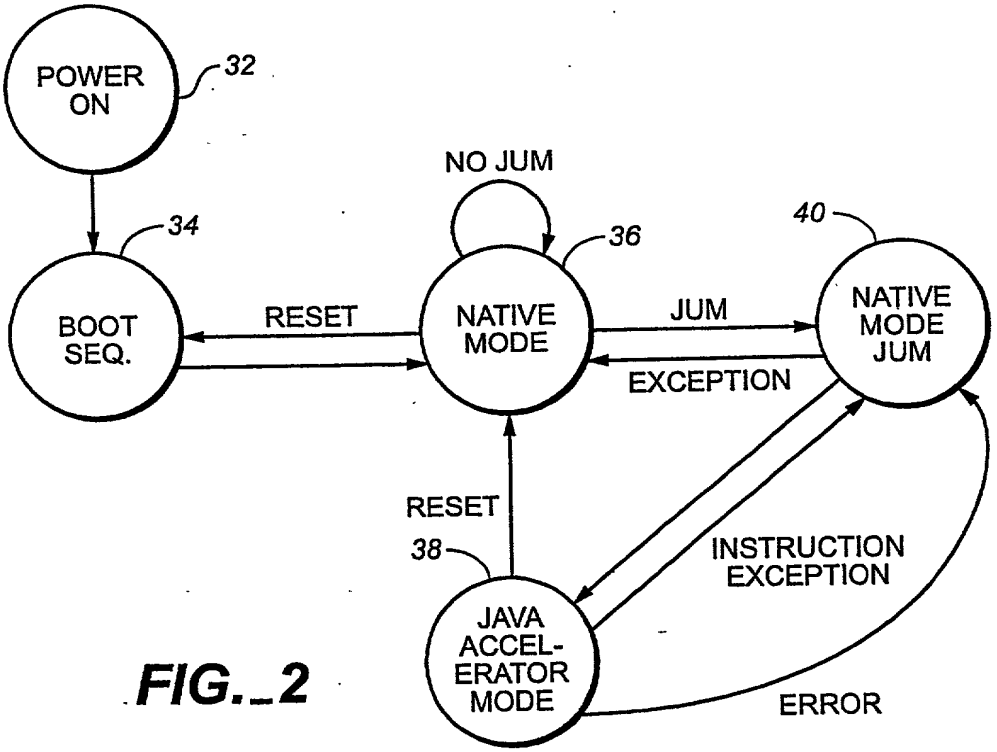
POWER ON
32
34
BOOT SEQ.
RESET
NO JUM
36
NATIVE MODE
JUM
EXCEPTION
40
NATIVE MODE JUM
RESET
38
JAVA ACCEL-ERATOR MODE
INSTRUCTION EXCEPTION
ERROR

FIG._2

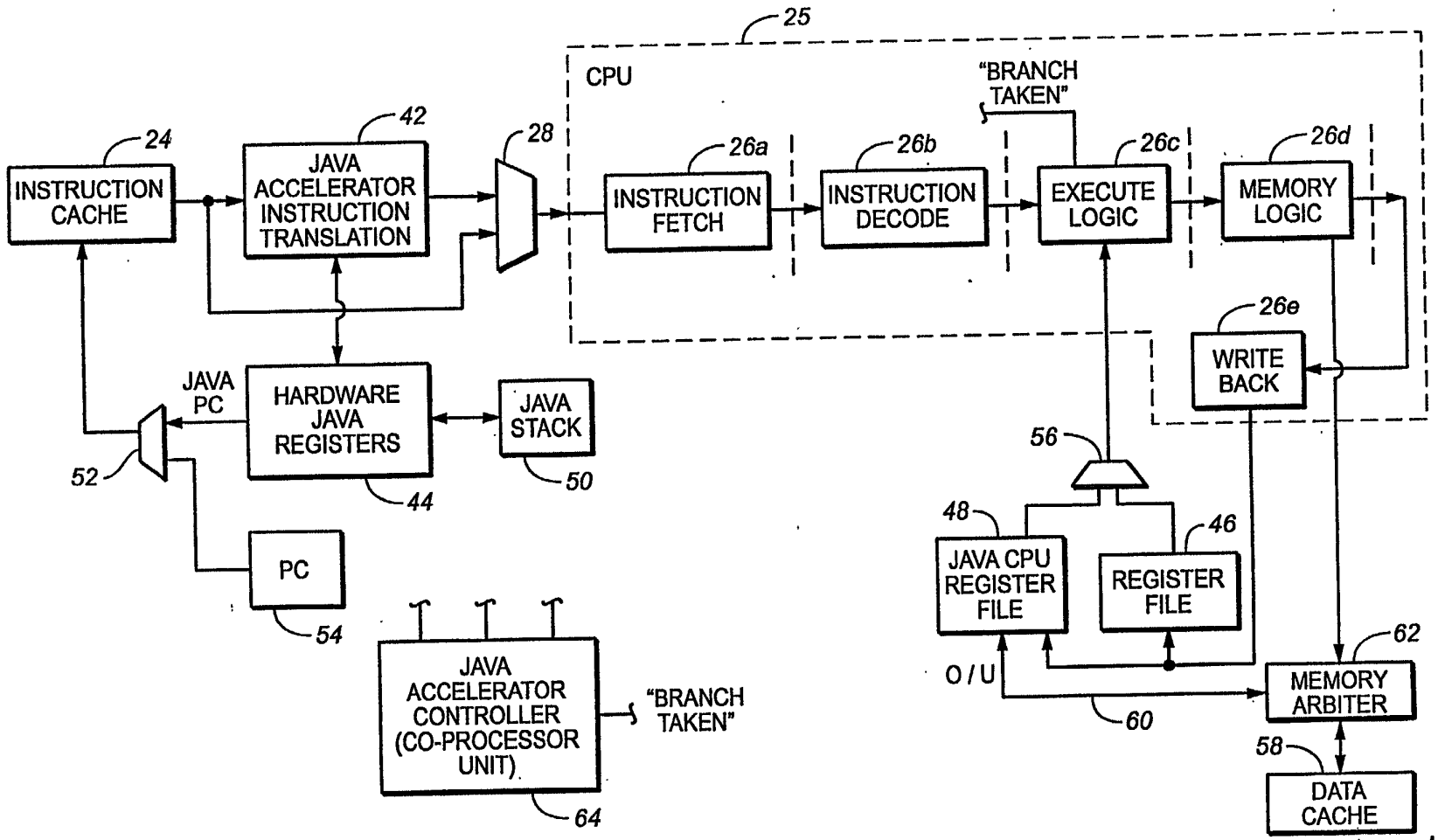

FIG._3

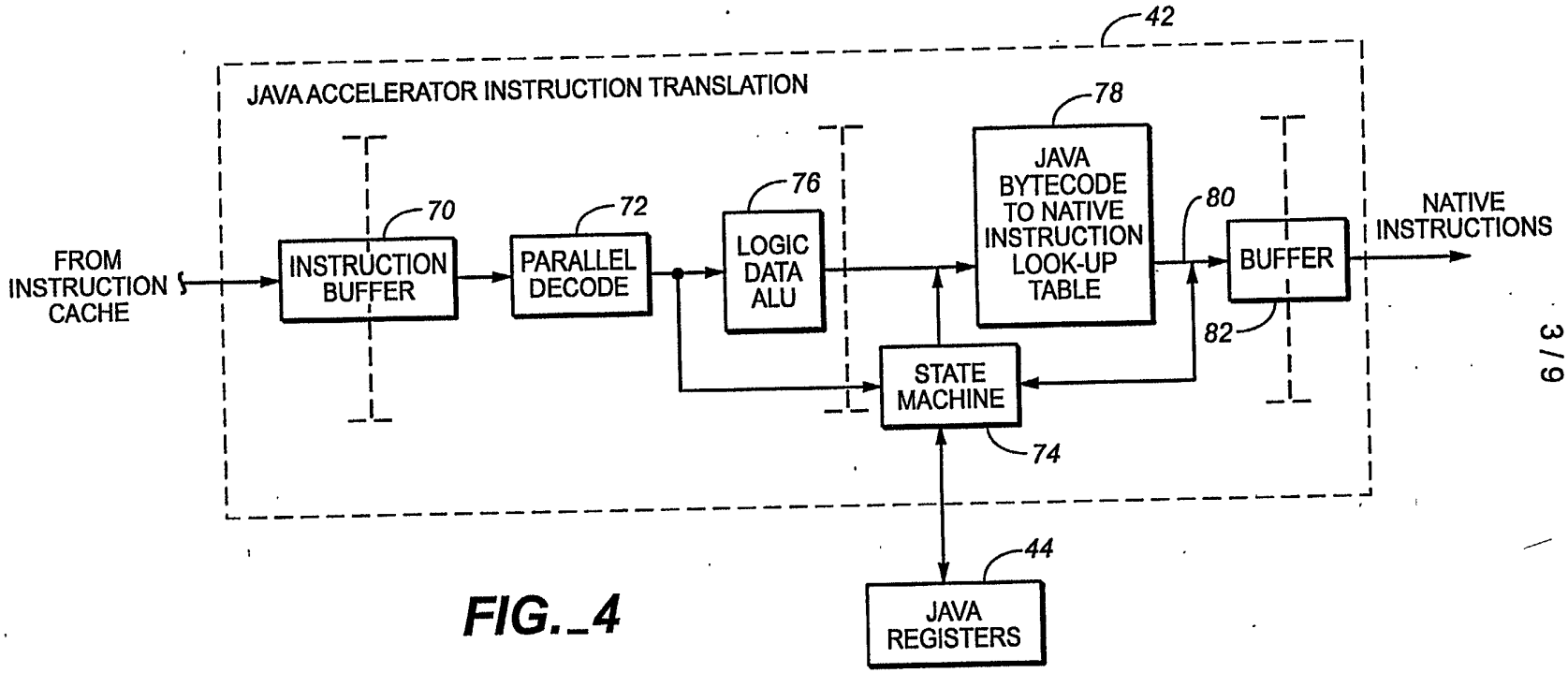
42
JAVA ACCELERATOR INSTRUCTION TRANSLATION
78
JAVA BYTECODE TO NATIVE INSTRUCTION LOOK-UP TABLE
76
LOGIC DATA ALU
70
INSTRUCTION BUFFER
72
PARALLEL DECODE
80
BUFFER
NATIVE INSTRUCTIONS
FROM INSTRUCTION CACHE
82
STATE MACHINE
74
44
JAVA REGISTERS

FIG._4

I. INSTRUCTION TRANSLATION

JAVA BYTECODE	⇨	NATIVE INSTRUCTION
iadd		ADD R1, R2

II. JAVA REGISTER

	⇨	
PC = VALUE A		PC = VALUE A + 1
OPTOP = VALUE B (R1)		OPTOP = VALUE B - 1 (R2)
VAR = VALUE C		VAR = VALUE C

III. JAVA CPU REGISTER FILE

		⇨		
	R0 0001			R0 0001
CONTAINS VALUE OF TOP OF OPERAND STACK →	R1 0150		NOT A VALID STACK VALUE →	R1 0150
	R2 1210		CONTAINS VALUE OF THE TOP OF OPERAND STACK →	R2 1360
	R3 0007			R3 0007
	R4 0005			R4 0005
	R5 0006			R5 0006
CONTAINS FIRST VARIABLE →	R6 1221			R6 1221
	R7 1361			R7 1361

IV. MEMORY

		⇨		
OPTOP = VALUE B →	- 0150			- 0150
(VALUE B - 1)	- 1210		OPTOP = VALUE B - 1	- 1360
	- 0007			- 0007
	- 0005			- 0005
	- 0006			- 0006
	- 0001			- 0001
	- 4427			- 4427
VAR = VALUE C	- 1221		VAR = VALUE C	- 1221
	- 1361			- 1361
	- 1101			- 1101

FIG._5

I. INSTRUCTION TRANSLATION

JAVA BYTECODE		NATIVE INSTRUCTION
iload_n iadd	⇨	ADD R6, R1

II. JAVA REGISTER

PC = VALUE A OPTOP = VALUE B (R1) VAR = VALUE C	⇨	PC = VALUE A + 2 OPTOP = VALUE B (R1) VAR = VALUE C

III. JAVA CPU REGISTER FILE

	R0	0001		R0	0001
CONTAINS VALUE OF TOP OF OPERAND STACK →	R1	0150	CONTAINS VALUE OF TOP OF STACK →	R1	1371
	R2	1210		R2	1210
	R3	0007		R3	0007
	R4	0005		R4	0005
	R5	0006		R5	0006
CONTAINS FIRST VARIABLE →	R6	1221	CONTAINS FIRST VARIABLE →	R6	1221
	R7	1361		R7	1361

IV. MEMORY

OPTOP = VALUE B →	-	0150	OPTOP = VALUE B	-	1371
	-	1210		-	1210
	-	0007		-	0007
	-	0005		-	0005
	-	0006		-	0006
	-	0001		-	0001
	-	4427		-	4427
VAR = VALUE C	-	1221	VAR = VALUE C	-	1221
	-	1361		-	1361
	-	1101		-	1101

FIG._6

Opcodes Mnemonic	Opcode xHH	Excep Gen
nop	0x00	
aconst_null	x01	
iconst_m1	x02	
iconst_n(0-5)	x03 - x08	
lconst_n(0-1)	x09 - x0a	
fconst_n(0-2)	x0c - x0d	
dconst_n(0-1)	x0e -x0f	
bipush	x10	
sipush	x11	
ldc	x12	y
ldc_w	x13	y
ldc2_w	x14	y
iload	x15	
lload	x16	
fload	x17	
dload	x18	
aload	x19	
iload_n(0-3)	x1a - x1d	
lload_n(0-3)	x1e - x21	
fload_n(0-3)	x22 - x25	
dload_n(0-3)	x26 - x29	
aload_n(0-3)	x2a - x2d	
iaload	x2e	
laload	x2f	
faload	x30	
daload	x31	
aaload	x32	
baload	x33	
caload	x34	
saload	x35	
istore	x36	
lstore	x37	
fstore	x38	
dstroe	x39	
astroe	x3a	
istore_n(0-3)	x3b - x3e	
lstore_n(0-3)	x3f - x42	
fstore_n(0-3)	x43 - x46	
dstore_n(0-3)	x47 - x4a	
astore_n(0-3)	x4b - x4e	
iastore	x4f	
lastore	x50	
fastroe	x51	
dastore	x52	
bastore	x53	
aastore	x54	
castroe	x55	
sastore	x56	

FIG._7A

pop	x57	
pop2	x58	
dup	x59	
dup_x1	x5a	
dup_x2	x5b	
dup2	x5c	
dup2_x1	x5d	
dup2_x2	x5e	
swap	x5f	
iadd	x60	
ladd	x61	
fadd	x62	y
dadd	x63	y
isub	x64	
lsub	x65	
fsub	x66	y
dsub	x67	y
imul	x68	
lmul	x69	
fmul	x6a	y
dmul	x6b	y
idiv	x6c	y
ldiv	x6d	y
fdiv	x6e	y
ddiv	x6f	y
irem	x70	y
lrem	x71	y
frem	x72	y
drem	x73	y
ineg	x74	
lneg	x75	
fneg	x76	y
dneg	x77	y
ishl	x78	
lshl	x79	
ishr	x7a	
lshr	x7b	
iushr	x7c	
lushr	x7d	
iand	x7e	
land	x7f	
ior	x80	
lor	x81	
ixor	x82	
lxor	x83	
iinc	x84	
i2l	x85	y
i2f	x86	y
i2d	x87	y
l2i	x88	y
l2f	x89	y
l2d	x8a	y

FIG._7B

f2i	x8b	y
f2l	x8c	y
f2d	x8d	y
d2i	x8e	y
d2l	x8f	y
d2f	x90	y
i2b	x91	
i2c	x92	
i2s	x93	
lcmp	x94	y
fcmpl	x95	y
fcmpg	x96	y
dcmpl	x97	y
dcmpg	x98	y
ifeq	x99	
ifne	x9a	
iflt	x9b	
ifge	x9c	
ifgt	x9d	
ifle	x9e	
if_icmpeq	x9f	
if_icmpne	xa0	
if_icmplt	xa1	
if_acmpge	xa2	
if_cmpgt	xa3	
if_icmple	xa4	
if_acmpeq	xa5	
if_acmpne	xa6	
goto	xa7	
jsr	xa8	
ret	xa9	
tableswitch	xaa	y
lookupswitch	xab	y
ireturn	xac	
lreturn	xad	
freturn	xae	
dreturn	xaf	
areturn	xb0	
return	xb1	
getstatic	xb2	y
putstatic	xb3	y
getfield	xb4	y
putfield	xb5	y
invokevirtual	xb6	y
invokespecial	xb7	y
invokestatic	xb8	y
invokeinterface	xb9	y
xxunusedxxx	xba	y
new	xbb	y
newarray	xbc	y
anewarray	xbd	y
arraylength	xbe	y

FIG._7C

athrow	xbf	y
checkcast	xco	y
instanceof	xc1	y
monitorenter	xc2	y
monitorexit	xc3	y
wide	xc4	y
multianewarray	xc5	y
ifnull	xc6	y
ifnonnull	xc7	y
goto_w	xc8	
jsr_w	xc9	
ldc_quick	xcb	y
ldc_w_quick	xcc	y
ldc2_w_quick	xcd	y
getfield_quick	xce	y
putfield_quick	xcf	y
getfield2_quick	xd0	y
putfield2_quick	xd1	y
getstatic_quick	xd2	y
putstatic_quick	xd3	y
gtestatic2_quick	xd4	y
putstatic2_quick	xd5	y
invokevirtual_quick	xd6	y
invokenonvirtual_quick	xd7	y
invokesuper_quick	xd8	y
invokestatic_quick	xd9	y
invokeinterface_quick	xda	y
invokevirtualobject_quick	xdb	y
new_quick	xdc	y
anewarray_quick	xde	y
multinewarray_quick	xdf	y
checkcast_quick	xe0	y
instanceof_quick	xe1	y
invokevirtual_quick_w	xe2	y
getfield_quick_w	xe3	y
putfield_quick_w	xe4	y
breakpoint	xca	y
impdep1	xfe	y
impdep2	xff	y

FIG._7D